



This Presentation Courtesy of the
International SOA Symposium
 October 7-8, 2008 Amsterdam Arena
www.soasymposium.com
info@soasymposium.com

Founding Sponsors



Platinum Sponsors



Gold Sponsors




Silver Sponsors



REST or SOAP?

Radovan Janecek
 Chief Architect, BTO, HP Software
 October 2008



© 2008 Hewlett-Packard Development Company, L.P.
 The information contained herein is subject to change without notice

Eight Years of SOA Wins and Mistakes

- Co-founded Systinet (2000)
 - Web Services stacks in C++ and Java
 - Service Registry
 - SOA Governance
- Led SOA Center in Mercury/HP (2006)
 - SOA Governance, Quality, Management
- BTO Architecture (2008)
 - Service and Data Models
 - Integration strategy (SOA based)



To Remember

- SOA is GOOD as it SIMPLIFIES big initiatives
 - Business Service Management
 - Business Service Automation
 - Service Portfolio Management
- SOA encompasses both REST and WS-^{*}
 - <http://www.google.com/search> is a service!
- REST vs WS-^{*} is not black&white decision



4 21 October 2008

REST vs. SOAP?

- REST is architecture style while SOAP is data envelope

- REST vs. WSDL or WSA is more research view
- HTTP vs. WSDL or WSA is more practical view

- Uniform interface vs arbitrary interface
- Every Time Based vs. Every Operation procedure

- So: the title of my presentation is nonsense!

5 21 October 2008



Let's be Even More Practical!

- RSS, Atom, APP, JSON, HTTP, OpenSearch, ...
- ... vs SOAP, WS-Addressing, WS-ReliableMessaging, ...

- Email clients, browsers, feed readers, syndicators, mashups, commandline tools, Office tools, GUI widgets, SaaS

!!!ALL OUT OF BOX!!!

6 21 October 2008



Key Considerations

Requirements

- Enterprise readiness
- Integration use cases
- User-facing use cases

Scalability

- # of different integrated systems
- # of users
- # of changes (future evolution, flexibility)
- Amount of data

Legacy

- Existing APIs
- Openness of existing systems
- Reusability

Organization

- Lack of time
- Lack of budget
- Skills and experience
- Governance processes maturity



Let's look into the details

CONSIDERATIONS



Skillset

- EAI mindset and experience is usually strong
 - CORBA
 - J2EE (EJBs)
- API oriented design is what we master
- Reinventing architectures
 - My architecture is bigger than yours!
- WSA leads to faster results (from service POV)
- RESTful services are more difficult to build



9 21 October 2008



Use Cases

- User facing use cases prevail
- M2M use cases usually simple
 - Data transfer
 - Simple orchestration
 - SUPPORT user facing use cases anyway
- Pull model, data services, feeds well proven in M2M on the web
- REST wins: wider adoption, few difficult use cases

10 21 October 2008



Enterprise Readiness

- Security, Failover, Reliability, Monitoring & Management
- Orchestration, 'business processes'

YOU NEED A MIDDLEWARE TO DO THIS!



11 21 October 2008



NO

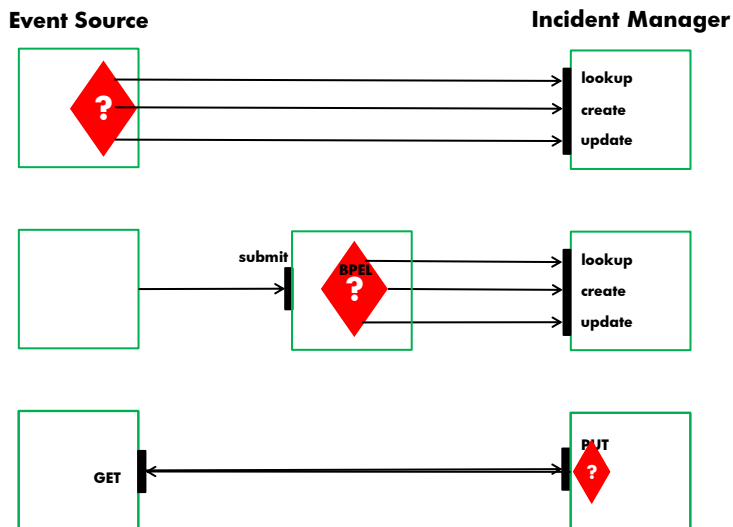
Good Service is Enterprise Ready!

- It secures itself
- It provides idempotent operations and ways to compensate
- It offers management interfaces
- Hide middleware behind your façade if you think you can't live without it
- REST wins: it teaches you how to do better without middleware

13 21 October 2008



Four Ways for One Use Case



14 21 October 2008



Enterprise Readiness Summary

- It is non-consideration
- Both WS-* and REST services are enterprise ready
- Both can be implemented in very bad way

- This is rather relevant to Maturity and skill set assessment

15 21 October 2008



Reusability

- Why to use SOAP for list of PurchaseOrders for M2M and Atom/JSON feed for the same on GUI level?

- Why to use collect data via SOAP if Atom offers better metadata and is far more adopted?
 - And instead of BPEL you can use mashup?

- REST wins: reusability is about standards and their real adoption in variety of different use cases

16 21 October 2008



Exposing Legacy

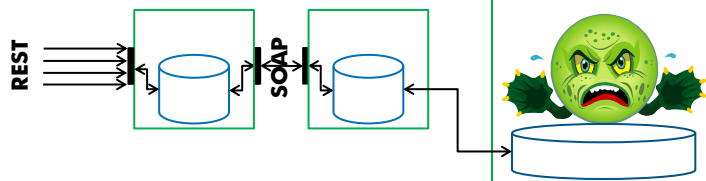
- Closed legacy systems might require EAI layer
- EAI vendors support SOAP better
- Open systems usually expose non-uniform API
 - Direct path to WSDL
 - Much easier to use WSA than REST
 - Beware of the API granularity issue!
- WSA can often be better (cheaper) choice here
- My suggestion: hide WSA by REST in different tier



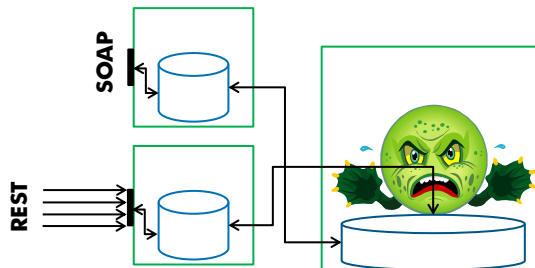
17 21 October 2008



Legacy Exposure Visual



"Impedance mismatch"
API
Objects vs RDBMS
Middleware relevant



18 21 October 2008



Scalability: # of Changes and Evolution

 20 / 5 = 4



Track status of BA 856 from London (LHR) to Prague (PRG)
2 Sep 2008 - On schedule
Departure: 3:00 PM, Arrival: 6:00 PM

Weather for San Francisco, CA - [Add to iGoogle](#)

16°C

Current: Clear
Wind: NW at 0 km/h
Humidity: 68%



26°C | 13°C 23°C | 13°C 23°C | 13°C 22°C | 13°C

[HPQ](#) - Hewlett-Packard Company (NYSE)

[Google Finance](#) [Yahoo Finance](#) [MSN Money](#) [AOL Finance](#) [CNN Money](#) [Reuters](#)



46.92 -0.41 (-0.87%) Aug 29 4:04pm ET

Open: 46.75 Volume: 0
High: 47.34 Avg Vol: 14,146,000
Low: 46.67 Mkt Cap: 115.71B

[Disclaimer](#)

After Hours: 47.32 +0.40 (0.85%) Aug 29 7:49pm ET



5 meters = 196.850394 inches



1:34pm Tuesday (CEST) - Time in Amsterdam, Netherlands
Amsterdam, New York 7:34am EDT

How many different APIs?
How many changes into client code?
Impact on existing clients?

- <http://www.google.com/search?q=20/5>
- <http://www.google.com/search?q=ba856>
- <http://www.google.com/search?q=5m+in+inches>
- <http://www.google.com/search?q=weather+San+Francisco>

19 21 October 2008

REST wins



Scalability: # of Systems

- Data syndication
 - Warehouse vs mashups
- GUI mashups
 - WS-Portlets vs JSON and widgets
- Complex orchestration
 - Process vs pub/sub
 - Often academic issue
- You cannot practically scale with WSA



20 21 October 2008



WSA vs REST Summary

- Service development cost: **WSA design is simpler**
- Development skills: **WSA more 'traditional'**
- Clarity and learning materials: **REST is the best documented**
- Scalability: **Caches, constraints**
- Simplicity of architecture: **Constraints: you can assume**
- Reusability of services: **Learn by example, standards**
- Standards adoption: **No discussion**
- Exposing legacy:
- Middleware: **YAGNI**

21 21 October 2008



Revised Summary

- It's not RED or GREEN decision
- Both REST and WSA will co-exist for a long time
- Many practical reasons for WSA might exist
 - Existing API, Skills, investments
- Steer the direction towards to REST



22 21 October 2008



On WSA: Avoid Mistakes

- WSA implementations often do not deliver on loose coupling promise
 - Direct legacy API exposure to SOA is very risky
 - Fine grained vs coarse grained
 - WSDL-based code generation is bad practice
 - Fragile from evolution perspective
 - Forward compatibility is a big issue
 - Strong typing is tightly coupled
 - XML Schema validation is BAD!
- Short-term advantage versus future pain

EAI in angle brackets

23 21 October 2008



Q&A

THANK YOU

24 21 October 2008

